INFORMATION AND COMPUTATION 94, 232–249 (1991)

On Oblivious Branching Programs of Linear Length

MATTHIAS KRAUSE

Universität Dortmund, Fachbereich Informatik, PO-Box 50 05 00, 4600 Dortmund 50, Germany

AND

STEPHAN WAACK

Karl-Weierstrass-Institut für Mathematik, Mohrenstrasse 39, Berlin 1086, Germany

Input oblivious decision graphs of linear length are considered. Among other concerns the computational complexity of three graph accessibility problems and the word problem of the free group are investigated. Several exponential lower bounds are proved.

1. INTRODUCTION

One major goal of complexity theory is to seperate complexity classes such as $\mathbb{L}$ and $\mathbb{NL}$ or to prove their coincidence. As usual, $\mathbb{L}$ and $\mathbb{NL}$ denote the classes of all languages A which can be accepted by deterministic and nondeterministic logspace bounded Turing machines, respectively. The nonuniform counterparts $\mathscr{L}$ and $\mathscr{NL}$ are the languages for which there are a polynomial $p(n)$ and an advice $\alpha_n \in \{0, 1\}^*$, where $|\alpha_n| \leqslant p(n)$, such that a deterministic or nondeterministic Turing machine, resp., accepts $w \# \alpha_n$ within logspace, $|w| = n$ ($\#$ is an additional tape symbol) if and only if w belongs to A.

A *Σ-decision graph* (DG) T_n, for Σ a finite alphabet, is a directed acyclic graph with the following properties.

- It has exactly one source, i.e., a node with indegree 0.
- Every node has outdegree 0 or $|\Sigma|$.
- Sinks, i.e., nodes with outdegree 0 are labelled by 0 or 1.
- Branching nodes, i.e., nodes with outdegree $|\Sigma|$, are labelled i, for some $1 \leqslant i \leqslant n$, and the $|\Sigma|$ outgoing arcs are labelled by the element of Σ, where each $\sigma \in \Sigma$ occurs exactly once.

To every word $w_1 w_2 \cdots w_n = w \in \Sigma^n$ there corresponds a unique path p_w

0890-5401/91 $3.00

from the source to a sink (at a branching node labelled i, it chooses the arc labelled by w_i). The decision graph T_n decides a set $L^{(n)} \subseteq \Sigma^n$ iff for every $w \in \Sigma^n$ the sink at the end of the path p_w is labelled by $L^{(n)}(w)$. [Throughout this work we make no difference between $L^{(n)}$ and its characteristic function denoted by $L^{(n)}$, too.]

The *size* of a decision graph T_n, which we sometimes denote by $\mathrm{SIZE}(T_n)$, is the number of branching nodes of T_n.

To avoid confusions we shall use the terms *nodes* and *arcs* to refer to the elements of a decision graph. Multiarcs are allowed. When decision graphs are used to decide graph problems, these graphs will have *vertices* and *edges*.

A $\{0, 1\}$-decision graph is a *branching program* (BP). Branching programs compute Boolean functions. They have been studied more extensively than decision graphs over larger alphabets, although the latter ones are more adapted in many cases. But decision graphs, which are also called R-way branching programs, were studied for example in Alon and Maass (1986) and Borodin and Cook (1982).

The logarithm of the size of a smallest decision graph deciding a language is a lower bound on the space requirement for many reasonable sequential models of computation.

It is well-known that $\mathscr{P}_{\mathrm{BP}} = \mathscr{L} \cap 2^{\{0,1\}^*}$ and $\mathscr{P}_{\mathrm{OG}} = \mathscr{L}$, where $\mathscr{P}_{\mathrm{BP}}$ and $\mathscr{P}$DG are the classes of languages which can be accepted by branching programs and decision graphs, respectively, of polynomially bounded size. Efforts to prove lower bounds for branching programs are eventually aimed at separating $\mathbb{L}$ from other complexity classes.

Nonlinear lower bounds $(\Omega(n^2/(\log n)^2))$ have already been given by Nechiporuk (1966) (in the more general framework of contact schemes). In order to obtain larger lower bounds for branching programs and decision graphs, restricted models are considered. First we turn to decision graphs the multiplicity of reading of which is restricted.

A *read-k-times-only* decision graph is allowed to encounter each input variable at most k times along any computation path. It is called *real-time*, if, for every w the length of the computation path p_w is less than or equal to n. $\mathscr{P}_{\mathrm{DG}k}$ is defined to be the class of all formal languages which can be decided by a sequence of read-k-times-only decision graphs the size of which is polynomially bounded.

Read-once-only branching programs were studied by Wegener (1988), Zak (1984), Dunne (1989), Krause (1988), and Ajtai *et al.* (1986). Wegener, Zak, Dunne, and Krause gave $2^{\Omega(\sqrt{n})}$ lower bounds, whereas in Ajtai *et al.* (1986) a $2^{c \cdot n}$ lower bound was proved, for c approximately 10^{-13}. For example, the graph property studied by Hajnal, Turan, and Szemeredi in Ajtai *et al.* (1986) is "G has an even number of triangles." Zak investigated the property "G is a halfclique."

Clearly, the real-time model is more powerful than the read-once-only model. Again Zak (manuscript) proved a $2^{\Omega(\sqrt{n})}$ lower bound. Kriegel and Waack (1988) studied the real-time decision graph complexity of the Dyck language D_m^*. It is known that the membership problem for the Dyck language D_m^* is identical with the word problem of the free group of rank m. A $(2m)^{n/24}$ lower bound was obtained for real-time decision graphs, and a $2^{n/48}$ lower bound for the real-time branching program complexity of an encoding of D_m^*.

No superpolynomial lower bound is known even in the case of read-twice-only branching programs.

Another approach that recently gained popularity is proving lower bounds for levelled branching programs for which several additional constraints are imposed.

A decision graph is called *levelled* iff its set of nodes is partitioned in a sequence of pairwise disjoint sets (the levels) such that arcs go from each level to the next level only. The *width* of a levelled decision graph is the maximum number of nodes on any level. The *length* is the number of levels.

In Ajtai *et al.* (1986), an $\Omega(n \log n/\log\log n)$ lower bound was proved for the size of levelled branching programs the width of which is bounded by $(\log n)^{O(1)}$ for almost all symmetric Boolean functions and in particular for the following function. "The sum of the input variables is a quadratic residue mod p," where p is a prime between $p^{1/4}$ and $p^{1/3}$.

Alon and Maass (1986) studied oblivious decision graphs of bounded width. A decision graph is called *oblivious* (ODG) if it is levelled and the nodes of any level are labelled by one and the same input variable. In Alon and Maass (1986) among others the sequence equality function Q_{2n} is investigated. Q is defined over the 3-letter alphabet $\{0, 1, 2\}$. $Q_{2n}(a_1, a_2, ..., a_n, b_1, b_2, ..., b_n) = 1$ iff the sequence obtained from $a_1 a_2 \cdots a_n$ by omitting all occurrences of 2 coincides with the one obtained in the same way from $b_1 b_2 \cdots b_n$. It is shown that for any $1 \leqslant s \leqslant 1/4 \log n$, if the width of an oblivious decision graph computing Q_{2n} is at most $2^{n/2^s}$, then its length is $\Omega(n \cdot s)$.

Krause (1991) considered oblivious read-k-times-only branching programs with the additional restriction that the variables occur only blockwise and in each block in the same order (k^*-programs). He gave examples of functions which do not belong to $\mathscr{P}_{\mathrm{BP1}}$ but which can be computed by the help of polynomially bounded 2*-branching programs. In fact, a quadratic upper bound was given.

Further, an exponential lower bound for k^*-branching programs, where k is a fixed number, was proved for the following problem. Decide whether a given subset $Y \subseteq \mathbb{F}_n \times \mathbb{F}_n$ containes the graph of a polynomial over $\mathbb{F}_n$ of degree less than $n/3$, where n is assumed to be a prime number.

For $\log n \leqslant s(n) \leqslant n$, s nondecreasing, let $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(S(n))$ be the class of all formal languages over a finite alphabet which can be decided by a sequence of oblivious decision graphs of linear bounded length, and of $O(S(n))$ bounded size.

$\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(n^{O(1)})$ is denoted by $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$.

The investigations which are carried out in this work are motivated as follows.

(i) There is no superpolynomial lower bound known for read-k-times-only decision graphs, if $k \geqslant 2$. It is interesting whether this is possible when imposing further constraints.

(ii) The so called graph accessibility problems are well-known in complexity theory. Savitch (1977) proved that the usual graph accessibility problem (GAP) is $\mathbb{N}\mathbb{L}$-complete with respect to logspace reductions. In Meinel (1987) it is shown that GAP is $\mathscr{N}\mathscr{L}$-complete with respect to p-projection reductions, whereas the graphs accessibility problems GAP1 and GAPMON1 for directed graphs of outdegree one and directed monotone graphs of outdegree 1, respectively, (see Section 3) are proved to be $\mathscr{L}$-complete. The problem GAPMON1 should be easier than the problem GAP1. Is it possible to substantiate this?

(iii) Up to now there are essentially two types of models of restricted decision graphs for which superpolynomial lower bounds can be proved. These are the read-once-only model and the model of oblivious decision graphs of small length. The question is how these two models are related to each other.

We consider problems belonging to $\mathbb{L}$. The only exception is the word problems of one-relator groups. The results are the following.

(i) We prove exponential lower bounds for the graph accessibility problems GAP and GAP1 (Theorem 3.6), and for all word problems of finite group presentations for which there is a subset of the set of generators which is a basis for a free subgroup (Theorem 3.8).

(ii) In Section 2 we introduce the so called l-projection reductions. It turns out that

— $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{O(s(n))})$ is closed under l-projection reductions (Proposition 2.3);

— GAPMON1 belongs to $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$ (Proposition 2.4) and consequently because of Theorem 3.6 GAPMON1 is properly less than GAP1 with respect to l-projection reductions.

(iii) We prove the following results.

— $\mathscr{P}_{\mathrm{DG}1}$ is not contained in $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$. The sequence equality function Q_{2n} for which an exponential lower bound was proved in Alon and Maass

(1986) when input oblivious decision graphs are used does belong to $\mathscr{P}_{\mathrm{DG1}}$ (Proposition 2.7).

— $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$ is not contained in $\mathscr{P}_{\mathrm{DG1}}$. The function HALF-CLIQUE$_n$, which belongs to $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$ (Proposition 2.5), does not belong to $\mathscr{P}_{\mathrm{DG1}}$ (Zak, 1984).

— The union of $\mathscr{P}_{\mathrm{DG1}}$ and $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$ is properly contained in $\mathscr{L}$. The word problem of the free group for which there are exponential lower bounds for both models (see Kriegel and Waack (1988) and Theorem 3.8) belongs to $\mathbb{L}$. This result suggests that current techniques do not suffice to separate $\mathbb{L}$ from larger complexity classes.

2. Reducibility, and Upper Bounds

It is standard in complexity theory to introduce reducibility notions in order to compare the complexity of two given problems. In accordance with Skyum and Valiant (1981) we say that a mapping π_n: $\{y_1, y_2, ..., y_m\} \to \{x_1, \bar{x}_1, x_2, \bar{x}_2, ..., x_n, \bar{x}_n, 0, 1\}$ is a projection reduction from a set $A \subseteq \{0, 1\}^n$ to a set $B \subseteq \{0, 1\}^m$ iff

$$A(x_1, x_2, ..., x_n) = B(\pi_n(y_1), \pi_n(y_1), ..., \pi_n(y_m)).$$

Equivalently, this means that $A = (\pi_n^*)^{-1}(B)$, where

$$\pi_n^*: \{0, 1\}^n \to \{0, 1\}^m$$

is the canonical map resulting from π_n.

Let L and L' be two sets contained in $\{0, 1\}^*$. $\{\pi_n: \{y_1, y_2, ..., y_{p(n)}\} \to \{x_1, \bar{x}_1, x_2, \bar{x}_2, ..., x_n, \bar{x}_n, 0, 1\} \mid n \in \mathbb{N}\}$ is called a p-projection reduction from L to L', if for each $n \in \mathbb{N}$ π_n is a projection reduction from $L^{(n)}$ to $L'^{(p(n))}$, and if $p(n) = O(n^k)$, where k is a constant. Then we say that L is p-projection reducible to L'.

$\{\pi_n \mid n \in \mathbb{N}\}$ is called an l-projection reduction iff $p(n) = O(n)$.

For practical reasons we generalize the notion of a projection reduction to languages over an arbitrary alphabet.

Let Σ and Γ be finite alphabets, and let $A \subseteq \Sigma^n$ and $B \subseteq \Gamma^m$ be two sets. A projection reduction π_n from A to B is defined to be

$$\pi_n = \{\pi_{n,0}, \pi_{n,i} \mid i \in \mathscr{I}\}$$

such that the following conditions are fulfilled.

(i) $\pi_{n,0}$ is a map from $\{1, 2, ..., m\} \to \{1, 2, ..., n\} \sqcup \Gamma$, where for any two sets the binary operation symbol "$\square$" means the disjoint union.

(ii) The index set $\mathscr{I}$ is defined to be $\pi_{n,0}^{-1}(\{1, 2, ..., n\})$.

(iii) The local functions $\pi_{n,i}$, $i \in \mathscr{I}$, map Σ to Γ.

(iv) $A = (\pi_n^*)^{-1}(B)$, where $\pi_n^*: \Sigma^n \to \Gamma^m$ is defined as follows:

$$(\pi_n^*(w))(i) := \begin{cases} \pi_{n,i}(w(\pi_{n,0}(i))) & \text{if } i \in \mathscr{I} \\ \pi_{n,0}(i) & \text{otherwise.} \end{cases}$$

We agree that if w is a word over a finite alphabet, then $w(i)$ or w_i denotes the ith letter of w.

The proof of Lemma 2.1 can be viewed as an illustration of this kind of reduction.

We remark that the reducibility notion for arbitrary alphabets coincides with the usual one for Boolean functions, if we restrict ourselves to $\Sigma = \Gamma = \{0, 1\}$. Now we can define p-projection reductions and l-projection reductions for languages over arbitrary alphabets in the straightforward way.

Obviously, p-projection reducibility relation as well as l-projection reducibility relation are transitive. We are justified to say that two given languages are equivalent with respect to p-projection reductions or with respect to l-projection reductions.

Remember that a language L is complete for a complexity class $\mathbb{K}$ with respect to a reducibility notion iff $\mathbb{K}$ is closed under this reductions, $L \in \mathbb{K}$, and each language $K \in \mathbb{K}$ can be reduced to the language L.

One natural way to get l-projection reductions is to consider reductions via a balanced homomorphism.

A homomorphism ϕ: $\Sigma^* \to \Gamma^*$, where Σ and Γ are finite alphabets, is called *balanced* iff for all σ, $\sigma' \in \Sigma$ we have $|\phi(\sigma)| = |\phi(\sigma')| =: |\phi|$.

A language $L \subseteq \Sigma^*$ is called *bh-reducible* to a language$L' \subseteq \Gamma^*$ if there is a balanced homomorphism ϕ: $\Sigma^* \to \Gamma^*$ such that $L = \phi^{-1}(L')$.

We observe that $L^{(n)} = \phi^{-1}(L'^{(|\phi| \cdot n)})$.

LEMMA 2.1. *If $L \subseteq \Sigma^*$ is bh-reducible to $L' \subseteq \Gamma^*$, then L is l-projection reducible to L'.*

Proof. Let ϕ: $\Sigma^* \to \Gamma^*$ be a balanced homomorphism such that $\phi^{-1}(L') = L$. Let $l(n) := |\phi| \cdot n$. Define

$$\pi_{n,0}: \{1, 2, ..., l(n)\} \to \{1, 2, ..., n\} \qquad \text{by}$$

$$\pi_{n,0}(i) = r \qquad \text{iff} \qquad i = (r-1)\,|\phi| + k, \qquad \text{for some } 1 \leqslant k \leqslant |\phi|,$$

and define $\pi_{n,i}$: $\Sigma \to \Gamma$ by $\pi_{n,i}(\sigma) := (\phi(\sigma))(k)$, where $1 \leqslant k \leqslant |\phi|$, and $k = i \bmod |\phi|$. Since $\pi_n^* = \phi$, we have that $w \in L^{(n)}$ iff $\pi_n^*(w) \in L'^{(l(n))}$. ∎

We consider the *graph accessibility problems* for directed graphs. A directed graph $G=(V, E)$, where $V=\{v_1, ..., v_N\}$ is the set of vertices and E is the set of edges, is uniquely determined by its adjacency matrix

$$A(G)=(a_{i,j})_{1 \leqslant i,j \leqslant N, i \neq j}$$

with

$$a_{i,j}=a(i,j)=\begin{cases} 1 & \text{if } (v_i, v_j) \in E \\ 0 & \text{otherwise.} \end{cases}$$

We define GAP $:= \{\mathrm{GAP}_{N(N-1)}\colon \{0,1\}^{N(N-1)} \rightarrow \{0,1\} \mid N \in \mathbb{N}\}$ by

$$(a_{i,j})_{i,j} \xmapsto{\mathrm{GAP}_{N(N-1)}} \begin{cases} 1 & \text{if there is a path from vertex} \\ & v_1 \text{ to vertex } v_N \\ 0 & \text{otherwise,} \end{cases}$$

GAP1 $:= \{\mathrm{GAP1}_{N(N-1)}\colon \{0,1\}^{N(N-1)} \rightarrow \{0,1\} \mid N \in \mathbb{N}\}$ by

$$(a_{i,j})_{i,j} \xmapsto{\mathrm{GAP1}_{N(N-1)}} \begin{cases} 1 & \text{if there is a path from vertex } v_1 \\ & \text{to vertex } v_N \text{, and outdegree}(G)=1 \\ 0 & \text{otherwise,} \end{cases}$$

and GAPMON1 $:= \{\mathrm{GAPMON1}_{N(N-1)}\colon \{0,1\}^{N(N-1)} \rightarrow \{0,1\} \mid N \in \mathbb{N}\}$ by $\mathrm{GAPMON1}_{N(N-1)}(a_{i,j})=1$ iff $\mathrm{GAP1}_{N(N-1)}(a_{i,j})=1$, and $a_{i,j}=1$ implies $i<j$.

In Meinel (1987) the following theorem is proved.

THEOREM 2.2. (i) GAP1 *as well as* GAPMON1 *are complete for* $\mathscr{L}$ *with respect to p-projection reductions.*

(ii) GAP *is complete for* $\mathcal{N}\mathscr{L}$ *with respect to p-projection reductions.*

The proof of the following proposition is very easy and so is omitted.

PROPOSITION 2.3. *Let* $s\colon \mathbb{N} \rightarrow \mathbb{N}$ *be a nondecreasing function such that* $\log n \leqslant s(n) \leqslant n$, *and for each* $\varepsilon>0$ *there is a* $\delta>0$ *such that* $s(\varepsilon \cdot n) \leqslant \delta \cdot s(n)$. *Then the class* $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{O(s(n))})$ *is closed with respect to l-projection reductions.*

PROPOSITION 2.4. GAPMON1 *belongs to* $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$.

Proof. In order to show that GAMON1 belongs to $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$, we describe a one-way Turing machine which works in logspace and which decides the problem. We assume that the adjacency matrix $A(G)$ is written in lines on the input tape. The machine simply follows the directed path starting in node 1. If there occurs more than one coefficient in one row which equals one, then it will reject. It has only to store the index of the current vertex on the working tape. This can be done in logspace. ∎

Another problem is to check whether an undirected graph G has a halfclique. Let $V=\{v_1, v_2, ..., v_N\}$ be the set of vertices. G is a halfclique iff there is a subset $I \subseteq \{1, 2, ..., N\}$ of cardinality $N/2$ such that $\{v_i \mid i \in I\}$ is a clique, whereas the complement of this set in V is an isolated set of vertices in G. The graph G is uniquely determined by the upper half of its adjacency matrix $A(G)=(a_{i,j})_{1 \leq i<j \leq N}$, where

$$a_{i,j}=\begin{cases}1 & \text{if } (v_i, v_j) \text{ is an undirected edge} \\ 0 & \text{otherwise.}\end{cases}$$

Let us denote the above defined function by $\mathrm{HALFCLIQUE}_n$: $\{0,1\}^n \to \{0,1\}$, where $n=\binom{N}{2}$.

PROPOSITION 2.5. (i) *The function* $\mathrm{HALFCLIQUE}_n$ *does not belong to* $\mathscr{P}_{\mathrm{BP1}}$.

(ii) $\mathrm{HALFCLIQUE}_n$ *belongs to* $\mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}}$.

Proof. The first claim is due to Zak (1984). In fact he proved a $2^{N/3-o(N)}$ lower bound, where N was the number of vertices of the graph under consideration.

Let us turn to claim (ii). We refine a construction due to Wegener (1988). It is well-known that there is a read-once-only branching program $B(x_1, x_2, ..., x_m)$ of size $\binom{m+1}{2}$ on m variables $x_1, ..., x_m$ which has $m+1$ output nodes numbered $0, 1, ..., m$, such that all inputs with exactly i ones reach the ith output. We get a program $B'(x_1, x_2, ..., x_m)$ by identifying all output nodes numbered by an element of the set $\{i \mid i \neq 0,\ i \neq m/2-1\}$. Here m is assumed to be even. The resulting node is defined to be rejecting. The program for which furthermore the sink numbered $m/2-1$ is declared to be accepting, whereas the sink numbered 0 is also a rejecting one, denoted by $B''(x_1, x_2, ..., x_m)$.

Now let us construct an input oblivious branching program of linear length for $\mathrm{HALFCLIQUE}_n$, $n=\binom{N}{2}$. Define

$$T_l := B'(x_{1,2}, x_{1,3}, ..., x_{1,l}, x_{l,l+1}, x_{l,l+2}, ..., x_{l,n}).$$

Then take a program $B''(y_1, y_2, ..., y_N)$ and replace all nodes of the lth level by the program T_l. There output 0 of T_l becomes the source of the arc of T_l labelled 0, output $N/2-1$ becomes the source of the arc labelled

1, and the third output of T_l is a rejecting one. Formally, of course, it is not allowed in an oblivious program to reject in a level which is not the last one. But we can easily overcome this problem by asking dummy questions.

The resulting program classifies correctly, since a graph G is a halfclique iff the cardinality of vertices having outdegree $N/2-1$ is $N/2$ whereas all other vertices have degree 0. ∎

In Ajtai *et al.* (1986) the notion of an *eraser Turing machine* is introduced. That machine has a special read-once-only input tape. By means of an indexing tape, the machine decides in the course of the computation in what order to read the input. After one input cell has been read, it is erased, and the machine will never ask for it again. In order to relate eraser Turing machines to decision graphs, we need the concept of *nonuniform eraser Turing machines*. These machines are allowed to use a polynomially restricted *advice* $\alpha(|w|)$, where $\alpha\colon \mathbb{N} \to \{0,1\}^*$, to decide the input w. The following theorem is proved in Meinel, Krause, and Waack (1988).

THEOREM 2.6. *The class $\mathscr{P}_{\mathrm{DG1}}$ equals the class of all languages which can be accepted by a* log *n-space bounded nonuniform eraser Turing machine.*

We use that theorem to prove the following

PROPOSITION 2.7. (i) *The sequence equality function Q_{2n} belongs to $\mathscr{P}_{\mathrm{DG1}}$.*

(ii) *The sequence equality function Q_{2n} does not belong to the class* $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{o(n)})$.

Hence $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{o(n)}) \subset \mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{O(n)})$.

Proof. Claim (ii) follows directly from the result of Alon and Maass (1980) which has already been quoted. Let us turn to claim (i). We describe a nonuniform logspace-bounded eraser Turing machine. For any input of even length $2n$ the advice is an encoding of the number n. All inputs of odd length are advised to be rejected.

The machine simulates two counters. The contents of counter one is the index of the input cell of the left part the machine will read next. Counter one is diminished by one after having read the associated cell.

The same is valid for counter two and the right part of the input. The only exception is, that the contents of counter two is increased by one after having read the input cell the index of which was stored.

Initially counter one equals n, and counter two equals $n+1$. The machine initializes them by the help of the advice. Then the computation is devided into at most n global steps.

During any global step the machine works as follows. It reads on the left part of the input as long as it will find a "0", a "1" or the left end marker of the input tape. Then it will turn to the right, and it will do the same. The input is rejected, if it will find another symbol than it found on the left side.

If no further global step is possible, and it has not yet rejected, the input is accepted. ▮

Now let us turn to word problems. First we consider free groups. Let $A = \{a_1, a_2, ..., a_m\}$, $m \geq 2$. Assume that $\langle A \rangle$ is the *free group on* A. The integer m is called *the rank* of the group. Then each element of $\langle A \rangle$ can be represented as a word over the alphabet $\underline{A} := A \,\square\, \{a_1^{-1}, a_2^{-1}, ..., a_m^{-1}\}$.

Given two words w_1 and w_2 over $\underline{A}$. It is well-known tht w_1 is *freely equal* to w_2, i.e., w_1 and w_2 define one and the same element in $\langle A \rangle$, iff w_1 can be transformed into w_2 by a finite sequence of the following rules: (i) replace $a_i a_i^{-1}$ by 1; (ii) replace $a_i^{-1} a_i$ by 1; (iii) the inverse of (i); (iv) the inverse of (ii), where 1 denotes the empty word.

A word w is called *freely reduced* iff neither rule (i) nor rule (ii) can be applied to w. Obviously, each group element of $\langle A \rangle$ has a *unique reduced representation over* $\underline{A}$.

A word is called *cyclically reduced* iff w is reduced and moreover the first letter of w is not equal to the inverse of the last letter.

In general, a group G is called *finitely presented* iff there are a finite set $A = \{a_1, a_2, ..., a_m\}$ and a finite set of reduced words $R = \{r_1, r_2, ..., r_s\}$ such that $G \cong \langle A \rangle / \mathrm{cl}(R)$, where $\mathrm{cl}(R)$ denotes the smallest normal subgroup containing R. We say that G has the finite presentation $\langle a_1, a_2, ..., a_m; R \rangle$ and we also write $G = \langle A; R \rangle$.

The *word problem* of $\langle a_1, a_2, ..., a_m; R \rangle$ is the language $W(\langle a_1, a_2, ..., a_m; R \rangle) := \{w \in \underline{A}^* \mid w = 1 \text{ in } G\}$. Let us denote $W^{(n)}(\langle a_1, a_2, ..., a_m; R \rangle) := W(\langle a_1, a_2, ..., a_m; R \rangle) \cap \underline{A}^n$.

Let us consider the following two examples. First we have a look at the presentation $\langle a_1, a_2, ..., a_m, \mathbf{1}; \mathbf{1} \rangle$. It is trivial that this is a presentation of the free group of rank m.

Second consider $\langle a_1, a_2, ..., a_m; r \rangle$, where r is a cyclically reduced word. Groups which can be represented in such a way are called *one-relator groups*. Two basic theorems on one-relator groups are the *Freiheitssatz* and the solvability of the word problem. These results were obtained by Magnus in the early 1930's (see Magnus (1930, 1932).)

THEOREM 2.8 (Freiheitssatz). *Let $G = \langle a_1, a_2, ..., a_m, ...; r \rangle$, where r is cyclically reduced. If L is a subset of $\{a_1, a_2, ..., a_m, ...\}$ which omits a generator occuring in r, the subgroup M generated by L is freely generated by L.*

Now we are prepared to prove the following.

LEMMA 2.9. (i) *The word problem* $W(\langle a_1, a_2, \ldots, a_m, \mathbf{1}; \mathbf{1}\rangle)$ *is bh-reducible to the word problem* $W(\langle a_1, a_2, \ldots, a_m\rangle)$.

(ii) *Let* $\langle a_1, a_2, \ldots, a_m; r\rangle$ *be a one-relator group. Then the word problem of* $\langle b_1, b_2, \ldots, b_m\rangle$ *is bh-reducible to the word problem of* $\langle a_1, a_2, \ldots, a_{m+1}; r\rangle$.

(iii) *Let* $G = \langle a_1, a_2, \ldots, a_m; R\rangle$ *be a group presentation. Let* $L \subseteq \{a_1, a_2, \ldots, a_m\}$ *such that* L *is a basis of a free subgroup of* G. *Then the word problem of the free group of rank* $|L|$ *is bh-reducible to* $W(\langle a_1, a_2, \ldots, a_m; R\rangle)$.

Proof. We define

$$\phi\colon \{a_1, a_2, \ldots, a_m, \mathbf{1}\}^* \to \{a_1, a_2, \ldots, a_m\}^*$$

by

$$\phi(a_i) := a_i a_i, \qquad i = 1, 2, \ldots, m$$

$$\phi(a_i^{-1}) := a_i^{-1} a_i^{-1}, \qquad i = 1, 2, \ldots, m$$

$$\phi(\mathbf{1}) := \phi(\mathbf{1}^{-1}) := a_1 a_1^{-1}.$$

This ϕ defines a group monomorphism. Hence claim (i) is proved.

Claim (ii) follows directly from the *Freiheitssatz.* Claim (iii) is obvious. ∎

Remark. The symbol "**1**" equals the unit of the free group. It is usually represented by the empty word. But it is useful in the proof of 3.7 to have this redundant generator for technical reasons.

3. LOWER BOUNDS

Put $[n] = \{1, 2, \ldots, n\}$ and let $\mathfrak{n} := (y_1, y_2, \ldots, y_r)$ be a sequence of elements of $[n]$. Let Z_1 and Z_2 be two disjoint substs of $[n]$.

We say that a $\{Z_1, Z_2\}$-alternation occurs at index i in the sequence $\mathfrak{n}$ iff the following conditions are satisfied.

(i) y_i belongs to $Z_1 \cup Z_2$.

(ii) There is a $k > i$ such that $y_k \in Z_1 \cup Z_2$.

(iii) $y_i \in Z_1$ iff $y_{\varkappa(i)} \in Z_2$, where $\varkappa(i) := \min\{k \mid k > i, y_k \in Z_1 \cup Z_2\}$.

The number of indices i at which there occurs a $\{Z_1, Z_2\}$-alternation is called the *alternation length of* $\mathfrak{n}$ *with respect to* $\{Z_1, Z_2\}$.

The following lemma is a straightforward consequence of a Ramsey-theoretic lemma due to Alon and Maass (1986).

LEMMA 3.1. *Assume that in the sequence* $\mathfrak{n}$ *each* $a \in [n]$ *appears at most* k *times. Then for any preassigned partition* $[n] = X_1 \sqcup X_2$ *of* $[n]$ *into two disjoint sets there are two subsets* $Y_i \subseteq X_i$, $i = 1, 2$, *such that*

— $|Y_i| \geqslant |X_i| \cdot 2^{-(2k-1)}$, $i = 1, 2$,

— *the alternation length of* $\mathfrak{n}$ *with respect to* $\{Y_1, Y_2\}$ *is less than or equal to* $2k$.

We associate with each input oblivious decision graph of length λ a sequence $\mathfrak{n} = (y_1, y_2, ..., y_\lambda)$ of indices, where y_i is the number with which the nodes of the ith level are labelled. $\mathfrak{n}$ is called the *index sequence* of the decision graph.

We need one technical notion. Let c_1, c_2: $[n]^{\supset} \to \Sigma$ be partial assignments such that c_1 and c_2 coincide on $\mathrm{dom}(c_1) \cap \mathrm{dom}(c_2)$. ($\mathrm{dom}(c) = \{i \mid c(i)$ is defined$\}$.) Define the union $c_1 \vee c_2$ as follows:

$$(c_1 \vee c_2)(i) = \begin{cases} c_1(i) & \text{if } \ i \in \mathrm{dom}(c_1) \\ c_2(i) & \text{if } \ i \in \mathrm{dom}(c_2) \\ \text{not defined} & \text{otherwise.} \end{cases}$$

DEFINITION. Let Z_1 and Z_2 be two disjoint subsets of $[n]$, let s_0 be a partial assignment, $\mathrm{dom}\, s_0 = [n] - Z_1 \cup Z_2$, and let $S_i \subseteq \{c \mid \mathrm{dom}\, c = Z_i\}$, $i = 1, 2$. Assume φ: $S_1 \to S_2$ to be a bijection. Further let $L^{(n)} \subseteq \Sigma^n$. The set $S = \{s_0 \vee s_1 \vee \varphi(s_1) \mid s_1 \in S_1\}$ is defined to be a *sheaf* in $L^{(n)}$, if and only if for all $s_1 \in S_1$ and all $s_2 \in S_2$ it holds that $s_0 \vee s_1 \vee s_2$ belongs to $L^{(n)}$ iff $s_2 = \phi(s_1)$.

$\{Z_1, Z_2\}$ is called the *support* of the sheaf S. The number $\log_2 |S|$ is called the *thickness* of the sheaf S.

In this work we often use sheaves in the context of the following lemma. The proof is pure routine and so is omitted. Informally speaking it claims that if palindromes are reducible to a language, then that language contains a sheaf.

LEMMA 3.2. *Let* Z_1 *and* Z_2 *be two disjoint subsets of the set of indices* $[n]$, *and let* $L^{(n)} \subseteq \Sigma^n$. *Assume that we are given a projection reduction* $\pi_n = \{\pi_{n,0}, \pi_{n,i}\ (i \in \mathscr{I})\}$, *where*

$$\pi_{n,0}\colon \{1, 2, ..., n\} \to \{1, 2, ..., 2\tau\} \sqcup \Sigma,$$

$$\mathscr{I} = \pi_{n,0}^{-1}(\{1, 2, ..., 2\tau\}), \qquad \textit{and} \qquad \pi_{n,i}\colon \{0, 1\} \to \Sigma,$$

from $\mathrm{PAL}^{(2\tau)} := \{ww^R \mid w \in \{0, 1\}^\tau\}$ *to* $L^{(n)}$ *such that* $\pi^{-1}(\{1, 2, ..., \tau\}) = Z_1$ *and* $\pi^{-1}(\{\tau + 1, ..., 2\tau)\} = Z_2$, *or vice versa.*

Then $\pi_n^*(\mathrm{PAL}^{(2\tau)})$ *is a sheaf in* $L^{(n)}$ *of thickness* τ. $\{Z_1, Z_2\}$ *is the support of this sheaf.*

The following lemma supplies a lower bound for input oblivious decision graphs in terms of sheafs of the problems under consideration. Similar methods were developed in Ajtai *et al.* (1986), Alon and Maass (1986), and Krause (1991).

LEMMA 3.3. *Let T_n be an input oblivious decision graph of width ω and length λ deciding a set $L^{(n)}$. Let α be the alternation length of T_n with respect to $\{Z_1, Z_2\}$, where Z_1, and Z_2 are disjoint subsets of $[n]$.*

If S is a sheaf in $L^{(n)}$ of thickness τ with support $\{Z_1, Z_2\}$, then

$$\omega \geqslant 2^{\tau/\alpha}.$$

Proof. Let $\mathfrak{n} = (y_1, ..., y_\lambda)$ be the index sequence of T_n. By definition the alternation length of $\mathfrak{n}$ with respect to $\{Z_1, Z_2\}$ equals α. By the definition of a sheaf we know that $S = \{s_0 \vee s_1 \vee \varphi(s_1) \mid s_1 \in S_1\}$, where

— s_0 is a partial assignment defined on $[n] - Z_1 \cup Z_2$,
— $S_i \subseteq \{c \mid \mathrm{dom}\, c = Z_i\}$, $i = 1, 2$,
— φ: $S_1 \to S_2$ is a bijection.

Define $G_0 = S_1$. Let $L_{a(1)}, ..., L_{a(\alpha)}$ be those levels of T_n where a $\{Z_1, Z_2\}$-alternation occurs at index $a(i)$ in the sequence $\mathfrak{n}$. We inductively define a sequence $l_0, l_1, ..., l_{\alpha-1}$ of nodes, where l_0 is the source of the decision graph, and $l_i \in L_{a(i)}$ as follows.

Assume that $l_1, l_2, ..., l_{i-1}$, $i \geqslant 1$, are defined. Let G_{i-1} be the largest subset M of S_1 such that all words belonging to $\{s_0\} \vee M \vee \phi(M)$ define a computation path through $l_0, ..., l_{i-1}$. By induction we get $|G_{i-1}| \geqslant 2^\tau/\omega^{i-1}$. Then we define l_i to be the node to which at least $2^{2\tau}/\omega^{2i-1}$ many of these words lead. Now it follows that

$$\omega \geqslant |L_{a(\alpha)}| \geqslant |G_{\alpha-1}| \geqslant 2^\tau/\omega^{\alpha-1}, \qquad \text{and hence} \qquad \omega \geqslant 2^{\tau/\alpha}. \quad \blacksquare$$

The following theorem claims that the complexity of a language is high if it contains a sheaf in a rather general position.

THEOREM 3.4. *Let s: $\mathbb{N} \to \mathbb{N}$ be a nondecreasing function, $\log n \leqslant s(n) \leqslant n$, and let $L \subseteq \Sigma^*$ be a language. Assume that for all ε, $0 < \varepsilon < 1/2$, there is a δ, $0 < \delta$, such that for infinitely many natural numbers n the following condition is fulfilled:*

There is a partition $I_1 \cup I_2 = [n]$ such that $|I_j| \geqslant \lfloor n/2 \rfloor$, $j = 1, 2$, and for any two subsets $Y_1 \subseteq I_1$, $Y_2 \subseteq I_2$, $|Y_i| \geqslant \varepsilon \cdot n$, there is a sheaf with support $\{Z_1, Z_2\}$ in $L^{(n)}$ of thickness greater than or equal to $\delta \cdot s(n)$, where $Z_i \subseteq Y_i$, $i = 1, 2$.

Then $L \notin \mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{o(n)})$.

Proof. Let $(T_n)_{n \in \mathbb{N}}$ be a sequence of input oblivious decision graphs of length $c \cdot n$, where c is fixed, and of width $\omega(n)$. Let us pick a natural number n for which the assumptions are fulfilled. Let $I_1 \cup I_2 = [n]$ be the partition. Let $\mathfrak{n}$ be the index sequence of T_n. Obviously, there are subsets $X_j \subseteq I_j$, $|X_j| \geqslant \lfloor n/4 \rfloor$, $j = 1, 2$, such that each $i \in X_1 \cup X_2$ occurs in $\mathfrak{n}$ at most $4c$ times. Then by Lemma 3.1 there are disjoint sets Y_1 and Y_2, $|Y_i| \geqslant n \cdot 2^{-8c}$, $i = 1, 2$, such that the alternation length of $\mathfrak{n}$ with respect to $\{Y_1, Y_2\}$ is bounded by $8c$, and $Y_i \subseteq X_i$.

By the assumptions there are a $\delta > 0$ and a sheaf with support $\{Z_1, Z_2\}$ in $L^{(n)}$ of thickness greater than $\delta \cdot s(n)$, where $Z_1 \subseteq Y_1$, and $Z_2 \subseteq Y_2$. Clearly, the alternation length of $\mathfrak{n}$ with respect to $\{Z_1, Z_2\}$ is also bounded by $8c$. By Lemma 3.3 it follows that

$$\log_2(\mathrm{SIZE}(T_n)) \geqslant (\delta/8c) \cdot s(n). \quad \blacksquare$$

LEMMA 3.5. *Let E be a subset of $\{(i, j) \mid 1 \leqslant i, j \leqslant N,\ i \neq j\}$, $|E| \geqslant \zeta \cdot N(N-1)$, where $\zeta \in (0, 1)$ is a real. Let $F \subseteq [n]$ be a "forbidden" set of numbers such that $1 \leqslant |F| \leqslant \tau \cdot N$, where τ is another constant, $0 < \tau < 1$, and $\zeta - 2\tau > 0$.*

Then there is a set $E' \subseteq \{1, 2, ..., N\}^3$ such that

(i) $|E'| \geqslant ((\zeta - \tau)/6)N - 1$

(ii) *$(h, i, j), (k, l, m) \in E'$ implies that* $|\{h, i, j, k, l, m\}| = 6$

(iii) *$(i, j, k) \in E'$ implies that $\{i, j, k\} \cap F = \varnothing$*

(iv) *$(i, j, k) \in E'$ implies that $(i, j) \in E$ and $(i, k) \in E$.*

Proof. We call a pair (i, j) incident to a number k iff $k \in \{i, j\}$. First we remove from E all pairs incident to a number belonging to F. We denote the remaining set of pairs by E''. Since at most $2\tau \cdot N(N-1)$ pairs are incident to a number from F, we get $|E''| \geqslant (\zeta - 2\tau) \cdot N(N-1)$.

At least $(\zeta - 2\tau) \cdot N$ pairs belonging to E'' have the same first component. We pick two of them. Thus we get a triple (i, j, k) such that $i \neq j \neq k \neq i$, $(i, j) \in E''$, $(i, k) \in E''$. (i, j, k) is the first element of the set E'. Next we add i, j, k to the forbidden set of numbers. Now the process iterates. We can proceed λ times in that way as long as

$$\frac{\zeta - 2 \cdot \tau}{6} \cdot N + \frac{2}{3} \geqslant \lambda. \quad \blacksquare$$

Now we are prepared to prove

THEOREM 3.6. *Both* GAP1 *and* GAP *do not belong to* $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{o(n^{1/2})})$.

Proof. The proofs for both assertions are identical. We shall settle down to GAP1. We shall carry out the proof on the basis of Theorem 3.4.

$\mathrm{GAP1}_{N(N-1)}$ is a Boolean function depending on $N(N-1)$ Boolean variables. The index set used is

$$\mathscr{I} = \{(i, j) \mid (i, j) \in \{1, ..., N\}^2, i \neq j\}.$$

Let Y_1 and Y_2 be two disjoint subsets of indices of $\mathscr{I}$ such that

$$|Y_i| \geqslant \zeta N(N-1), \qquad i = 1, 2.$$

Using Lemma 3.5 there are an $m = \Omega(N)$ and subsets Z_1 and Z_2 of Y_1 and Y_2, resp., such that

— $|Z_i| = 2m$, $i = 1, 2$,
— $(i, j) \in Z_1 \cup Z_2$ implies $\{1, N\} \cap \{i, j\}$ is empty,
— $|\{k \mid k$ is incident to an element (i, j) of $Z_1 \cup Z_2\}| = 7 \cdot m$.

Let

$$Z_1 = \{(a_i, b_i) \mid i = 1, 2, ..., m\} \cup \{(a_i, c_i) \mid i = 1, 2, ..., m\}$$
$$Z_2 = \{(d_i, e_i) \mid i = 1, 2, ..., m\} \cup \{(f_i, g_i) \mid i = 1, 2, ..., m\}$$

Now it remains to define a projection reduction

$$\pi\colon \{x_i \mid i \in \mathscr{I}\} \to \{y_1, \bar{y}_1, ..., y_{2m}, \bar{y}_{2m}, 0, 1\}$$

from $\mathrm{PAL}^{(2m)}$ to $\mathrm{GAP1}_{N(N-1)}$, where

$$\pi^{-1}(\{\bar{y}_1, y_1, ..., \bar{y}_m, y_m\}) = \{x_i \mid i \in Z_1\},$$
$$\pi^{-1}(\{y_{m+1}, \bar{y}_{m+1}, ..., y_{2m}, \bar{y}_{2m}\}) = \{x_i \mid i \in Z_2\}.$$

We write $x(i, j)$ instead of $x_{(i,j)}$. Define, for each $i \in \{1, ..., m\}$, $\pi(x(a_i, b_i)) := y_i$, $\pi(x(a_i, c_i)) := \bar{y}_i$, $\pi(x(d_i, e_i)) := y_{2m+1-i}$, and $\pi(x(f_i, g_i)) := \bar{y}_{2m+1-i}$. Moreover we set

$$\begin{aligned} 1 &= \pi(x(b_i, d_i)) = \pi(x(c_i, f_i)) = \pi(x(e_i, a_{i+1})) \\ &= \pi(x(g_i, a_{i+1})) = \pi(x(1, a_1)) = \pi(x(g_m, N)). \end{aligned}$$

Figure 1 illustrates this in the case of $m = 3$. The dotted arrows depend on the literals with which they are labelled. For example, the edge (a_i, c_i) exists iff $\bar{x}_i = 1$. All other edges are fixed. We observe that the triples (a_i, b_i, c_i) serve as switches. ▌

Now let us turn to word problems.

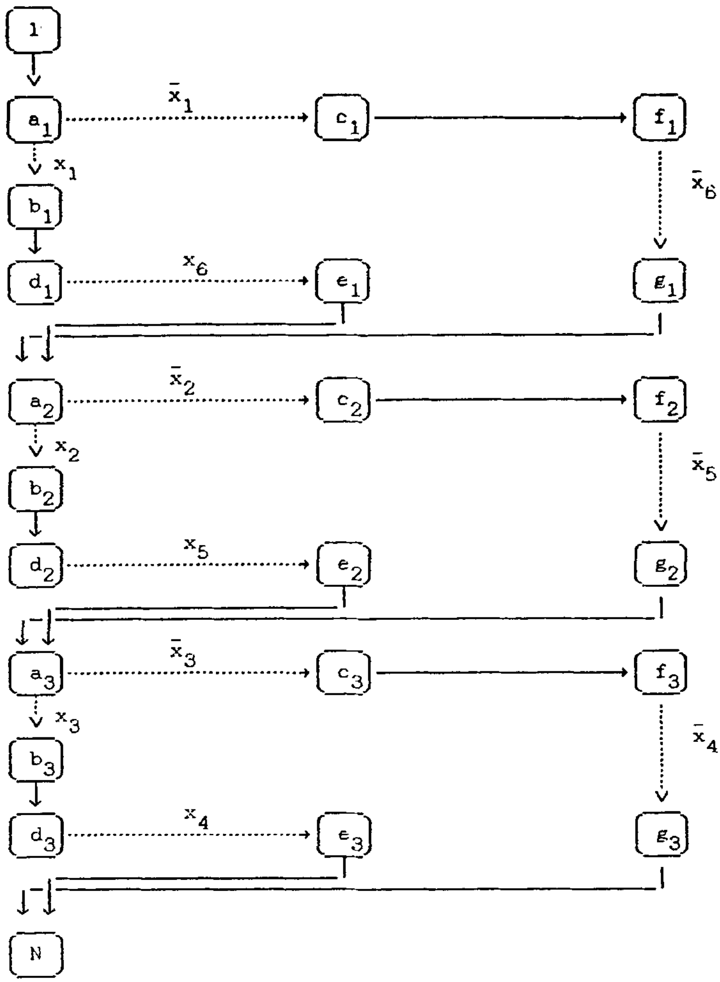

FIG. 1. $m=3$.

LEMMA 3.7. *The word problem of the presentation*

$$\langle a_1, a_2, ..., a_m, \mathbf{1}; \mathbf{1}\rangle$$

does not belong to $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{o(n)})$.

Proof. We have to apply Theorem 3.4. Let Y_1, Y_2 be two disjoint subsets of $[n]$, $|Y_i| \geqslant \varepsilon \cdot n$, $i=1, 2$. Let $Z_i \subseteq Y_i$ be the subsets such that $i \in Z_1$ and $j \in Z_2$ implies w.l.o.g. $i<j$. We know that $|Z_i| \geqslant (\varepsilon/2) \cdot n$, $i=1, 2$. Assume that $|Z_1| = |Z_2|$. We show that $\{Z_1, Z_2\}$ is the support of a sheaf in $W^n(\langle a_1, a_2, ..., a_m, \mathbf{1}; \mathbf{1}\rangle)$ of thickness $|Z_1| = |Z_2| \geqslant (\varepsilon/2) \cdot n =: n'$.

We put all input variables x_j, $j \notin Z_1 \cup Z_2$, to be **1**. Since **1** equals 1 in the group we have again a word problem of shorter length n'. We consider words of length n' over the alphabet $\{a_1, a_2, ..., a_m, a_1^{-1}, ..., a_m^{-1}\}$ of the type $uv^{-1} =: w(u, v)$, where $u, v \in \{a_1, a_2, ..., a_m\}^{n'/2}$. Obviously, $w(u, v) = 1$ iff $u \equiv v$. Now it is no problem to define a projection reduction from the palindromes to the word problem in the required way. ▮

If we combine the previous lemma with Lemma 2.9 and Proposition 2.3 we easily get

THEOREM 3.8. *The word problem of the presentation* $\langle a_1, a_2, ..., a_m \rangle$ *of the free group and of finitely generated one-relator groups* $\langle b_1, b_2, ..., b_m; r \rangle$ *do not belong to* $\mathrm{SIZE}_{\mathrm{DG}_{o,\mathrm{lin}}}(2^{o(n)})$.

COROLLARY 3.9. $\mathscr{P}_{\mathrm{DG1}} \cup \mathscr{P}_{\mathrm{DG}_{o,\mathrm{lin}}} \subset \mathscr{L}$.

Proof. The result follows from Theorem 3.8 and the well-known theorem due to Lipton and Zalcstein (1977) which states that the word problem of the free group is solvable in logspace. ▮

RECEIVED November 14, 1988; FINAL MANUSCRIPT RECEIVED March 13, 1990

REFERENCES

AJTAI, M., BABAI, L., HAJNAL, P., KOMOLOS, J., PUDLAK, P., ROEDEL, V., SEMEREDI, E., AND TURAN, G. (1986), Two lower bounds for branching programins, *in* "Proceedings, 18th ACM STOC," pp. 30–39.

ALON, N., AND MAASS, W. (1986), Meanders, Ramsay theory and lower bounds for branching programs, *in* "Proceedings, 27th IEEE—FOCS," 410–417.

BORODIN, A., AND COOK, S. (1982), A time-space tradeoff for sorting on a general sequential model of computation, *SIAM J. Comput.* **11** (1), 287–297.

DUNNE, P. E. (1985), Lower bounds for the complexity of 1-time-only branching programs, *in* "Proceedings, FCT," pp. 90–99, Lecture Notes in Computer Science, Vol. 199, Springer-Verlag, Berlin/New York.

KRAUSE, M. (1988), Exponential lower bounds on the complexity of real-time and local branching programs, *J. Inform. Process. Cybernet.* **24** 3.

KRAUSE, M. (1991), Lower bounds for depth-restricted branching programs, *Inform. Comput.* **91**, 1–14.

KRIEGEL, K., AND WAACK, S. (1988), Lower bounds on the complexity of real-time branching programs, *Informatique Théorique et Applications/Theoret. Informat. Appl.* **22** (4), 447–459.

LIPTON, R. J., AND ZALCSTEIN, Y. (1977), Word problems solvable in logspace, *J. Assoc. Comput. Mach.* **24** (3), 522–526.

MAGNUS, W. (1930), Über diskontinuierliche Gruppen mit einer definierenden Relation (Der Freiheitssatz), *J. Reine Angew. Math.* **163**, 141–165.

MAGNUS, W. (1932), Das Identitätsproblem für Gruppen mit einer definierenden Relation, *Math. Ann.* **106**, 195–307.

MEINEL, CH. (1987), The nonuniform complexity classes $\mathcal{NC}^1$, $\mathcal{L}$, and $\mathcal{NL}$, *J. Inform. Process. Cybernet.* **23** (10/11), 545–558.

MEINEL, CH., KRAUSE, M., AND WAACK, S. (1988), Separating the eraser Turing machine classes $\mathcal{L}_e$, $\mathcal{NL}_e$, $co\text{-}\mathcal{NL}_e$, and $\mathcal{P}_e$, *in* "Proceedings, MFCS," pp. 405–409, Lecture Notes in Computer Science, Vol. 324, Springer-Verlag, Berlin/New York.

NECHIPORUK, E. I. (1966), On a Boolean function, *Dokl. Akad. Nauk USSR* **169** (4), 765–766.

WEGENER, I. (1988), On the complexity of branching programs and decision trees for clique functions, *J. Assoc. Comput. Mach.* **35** (2), 461–471.

SAVITCH, W. (1977), Relationships between nondeterministic and deterministic tape complexities, *J. Comput. System Sci.* **4**, 177–192.

SKYUM, VALIANT, L. G. (1981), A complexity theory based on Boolean algebra, *in* "Proceedings, 22th FOCS," pp. 244–253.

ZAK, S. (1984), An exponential lower bound for one-time-only branching programs, *in* "Proceedings, MFCS," pp. 562–566, Lecture Notes in Computer Science, Vol. 176, Springer-Verlag, Berlin/New York.

ZAK, S. (manuscript), An exponential lower bound for real-time braching programs.